

## CLAIMS

1. A method for deriving a transform matrix, comprising:  
 deriving values for a  $2^m \times 2^m$  transform matrix using the following normalization constraints:

$$\left\{ \begin{array}{l} n_0 = norm \\ \sum_{i=0}^{2^{m-1}-1} n_{2^i+1}^2 = 2^{m-1} \cdot norm^2 \\ \sum_{i=0}^{2^{m-2}-1} n_{4^i+2}^2 = 2^{m-2} \cdot norm^2 \\ \sum_{i=0}^{2^{m-3}-1} n_{8^i+4}^2 = 2^{m-3} \cdot norm^2 \\ \vdots \\ n_{2^{m-1}} = norm \end{array} \right.$$

where, *norm* is an integer representing a normalization factor of the transform matrix; and selecting the *norm* that minimizes a DCT distortion function:

$$E_{2^m} = \frac{1}{2^m} \sum_{i=0}^{(2^m-1)} \sum_{\substack{j=0 \\ j \neq i}}^{(2^m-1)} \frac{|d_i(j)|}{|d_i(i)|}$$

where  $d_i = t_i \cdot DCT$ ,  $t_i$  is a base vector of the transform matrix, and DCT is a real Discrete Cosine Transform.

/

2. A method according to claim 1 wherein  $m = 16$  and the values of the transform matrix comprise the following:

$$T_{16} = \begin{bmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \\ t_{12} \\ t_{13} \\ t_{14} \\ t_{15} \end{bmatrix} = \begin{bmatrix} n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 \\ n_1 & n_3 & n_5 & n_7 & n_9 & n_{11} & n_{13} & n_{15} & -n_{15} & -n_{13} & -n_{11} & -n_9 & -n_7 & -n_5 & -n_3 & -n_1 \\ n_2 & n_6 & n_{10} & n_{14} & -n_{14} & -n_{10} & -n_6 & -n_2 & -n_2 & -n_6 & -n_{10} & -n_{14} & n_{14} & n_{10} & n_6 & n_2 \\ n_3 & n_9 & n_{15} & -n_{11} & -n_5 & -n_1 & -n_7 & -n_{13} & n_{13} & n_7 & n_1 & n_5 & n_{11} & -n_{15} & -n_9 & -n_3 \\ n_4 & n_{12} & -n_{12} & -n_4 & -n_4 & -n_{12} & n_{12} & n_4 & n_4 & n_{12} & -n_{12} & -n_4 & -n_4 & -n_{12} & n_{12} & n_4 \\ n_5 & n_{15} & -n_7 & -n_3 & -n_{13} & n_9 & n_1 & n_{11} & -n_{11} & -n_1 & -n_9 & n_{13} & n_3 & n_7 & -n_{15} & -n_5 \\ n_6 & -n_{14} & -n_2 & -n_{10} & n_{10} & n_2 & n_{14} & -n_6 & -n_6 & n_{14} & n_2 & n_{10} & -n_{10} & -n_2 & -n_{14} & n_6 \\ n_7 & -n_{11} & -n_3 & n_{15} & n_1 & n_{13} & -n_5 & -n_9 & n_9 & n_5 & -n_{13} & -n_1 & -n_{15} & n_3 & n_{11} & -n_7 \\ n_8 & -n_8 & -n_8 & n_8 & n_8 & -n_8 & -n_8 & n_8 & n_8 & -n_8 & -n_8 & n_8 & n_8 & -n_8 & -n_8 & n_8 \\ n_9 & -n_5 & -n_{13} & n_1 & -n_{15} & -n_3 & n_{11} & n_7 & -n_7 & -n_{11} & n_3 & n_{15} & -n_1 & n_{13} & n_5 & -n_9 \\ n_{10} & -n_2 & n_{14} & n_6 & -n_6 & -n_{14} & n_2 & -n_{10} & -n_{10} & n_2 & -n_{14} & -n_6 & n_6 & n_{14} & -n_2 & n_{10} \\ n_{11} & -n_1 & n_9 & n_{13} & -n_3 & n_7 & n_{15} & -n_5 & n_5 & -n_{15} & -n_7 & n_3 & -n_{13} & -n_9 & n_1 & -n_{11} \\ n_{12} & -n_4 & n_4 & -n_{12} & -n_{12} & n_4 & -n_4 & n_{12} & n_{12} & -n_4 & n_4 & -n_{12} & -n_{12} & n_4 & -n_4 & n_{12} \\ n_{13} & -n_7 & n_1 & -n_5 & n_{11} & n_{15} & -n_9 & n_3 & -n_3 & n_9 & -n_{15} & -n_{11} & n_5 & -n_1 & n_7 & -n_{13} \\ n_{14} & -n_{10} & n_6 & -n_2 & n_2 & -n_6 & n_{10} & -n_{14} & -n_{14} & n_{10} & -n_6 & n_2 & -n_2 & n_6 & -n_{10} & n_{14} \\ n_{15} & -n_{13} & n_{11} & -n_9 & n_7 & -n_5 & n_3 & -n_1 & n_1 & -n_3 & n_5 & -n_7 & n_9 & -n_{11} & n_{13} & -n_{15} \end{bmatrix}$$

where,

$$n_0 = 17, n_1 = 22, n_2 = 24, n_3 = 28, n_4 = 23, n_5 = 12, n_6 = 20, n_7 = 20, \\ n_8 = 17, n_9 = 12, n_{10} = 12, n_{11} = 16, n_{12} = 7, n_{13} = 8, n_{14} = 6, \text{ and } n_{15} = 6.$$

3. A method according to claim 1 including:  
receiving variable sized macroblocks of image data;  
selecting transform matrices corresponding to the variable sized macroblocks; and  
applying the selected transform matrices to the macroblocks.
4. A method according to claim 1 including using different  $4 \times 4$ ,  $8 \times 8$ , and  $16 \times 16$  transform matrices for Discrete Cosine Transforming different blocks of an image.

5. A method according to claim 1 including basing the constraints used for deriving the transform matrix on a Hadamard transform.

6. A system for processing data, comprising:  
a processor using a transform matrix:

$$T_{16} = \begin{bmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \\ t_{12} \\ t_{13} \\ t_{14} \\ t_{15} \end{bmatrix} = \begin{bmatrix} n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 & n_0 \\ n_1 & n_3 & n_5 & n_7 & n_9 & n_{11} & n_{13} & n_{15} & -n_{15} & -n_{13} & -n_{11} & -n_9 & -n_7 & -n_5 & -n_3 & -n_1 \\ n_2 & n_6 & n_{10} & n_{14} & -n_{14} & -n_{10} & -n_6 & -n_2 & -n_2 & -n_6 & -n_{10} & -n_{14} & n_{14} & n_{10} & n_6 & n_2 \\ n_3 & n_9 & n_{15} & -n_{11} & -n_5 & -n_1 & -n_7 & -n_{13} & n_{13} & n_7 & n_1 & n_5 & n_{11} & -n_{15} & -n_9 & -n_3 \\ n_4 & n_{12} & -n_{12} & -n_4 & -n_4 & -n_{12} & n_{12} & n_4 & n_4 & n_{12} & -n_{12} & -n_4 & -n_4 & -n_{12} & n_{12} & n_4 \\ n_5 & n_{15} & -n_7 & -n_3 & -n_{13} & n_9 & n_1 & n_{11} & -n_{11} & -n_1 & -n_9 & n_{13} & n_3 & n_7 & -n_{15} & -n_5 \\ n_6 & -n_{14} & -n_2 & -n_{10} & n_{10} & n_2 & n_{14} & -n_6 & -n_6 & n_{14} & n_2 & n_{10} & -n_{10} & -n_2 & -n_{14} & n_6 \\ n_7 & -n_{11} & -n_3 & n_{15} & n_1 & n_{13} & -n_5 & -n_9 & n_9 & n_5 & -n_{13} & -n_1 & -n_{15} & n_3 & n_{11} & -n_7 \\ n_8 & -n_8 & -n_8 & n_8 & n_8 & -n_8 & -n_8 & n_8 & n_8 & -n_8 & -n_8 & n_8 & n_8 & -n_8 & -n_8 & n_8 \\ n_9 & -n_5 & -n_{13} & n_1 & -n_{15} & -n_3 & n_{11} & n_7 & -n_7 & -n_{11} & n_3 & n_{15} & -n_1 & n_{13} & n_5 & -n_9 \\ n_{10} & -n_2 & n_{14} & n_6 & -n_6 & -n_{14} & n_2 & -n_{10} & -n_{10} & n_2 & -n_{14} & -n_6 & n_6 & n_{14} & -n_2 & n_{10} \\ n_{11} & -n_1 & n_9 & n_{13} & -n_3 & n_7 & n_{15} & -n_5 & n_5 & -n_{15} & -n_7 & n_3 & -n_{13} & -n_9 & n_1 & -n_{11} \\ n_{12} & -n_4 & n_4 & -n_{12} & -n_{12} & n_4 & -n_4 & n_{12} & n_{12} & -n_4 & n_4 & -n_{12} & -n_{12} & n_4 & -n_4 & n_{12} \\ n_{13} & -n_7 & n_1 & -n_5 & n_{11} & n_{15} & -n_9 & n_3 & -n_3 & n_9 & -n_{15} & -n_{11} & n_5 & -n_1 & n_7 & -n_{13} \\ n_{14} & -n_{10} & n_6 & -n_2 & n_2 & -n_6 & n_{10} & -n_{14} & -n_{14} & n_{10} & -n_6 & n_2 & -n_2 & n_6 & -n_{10} & n_{14} \\ n_{15} & -n_{13} & n_{11} & -n_9 & n_7 & -n_5 & n_3 & -n_1 & n_1 & -n_3 & n_5 & -n_7 & n_9 & -n_{11} & n_{13} & -n_{15} \end{bmatrix}$$

to transform the data, where:

$$n_0 = 17, n_1 = 22, n_2 = 24, n_3 = 28, n_4 = 23, n_5 = 12, n_6 = 20, n_7 = 20, \\ n_8 = 17, n_9 = 12, n_{10} = 12, n_{11} = 16, n_{12} = 7, n_{13} = 8, n_{14} = 6, \text{ and } n_{15} = 6.$$

7. A system according to claim 6 wherein the processor conducts a discrete cosine transform on the data according to the following:

$$C_{n \times m} = T_m \times B_{n \times m} \times T_n^T,$$

where  $B_{n \times m}$  is an image block of data with  $n$  pixels and  $m$  rows,  $T_n$  and  $T_m$  are the horizontal and vertical transform matrices of size  $n \times n$  and  $m \times m$ , respectively, and  $C_{n \times m}$  denotes the cosine transformed  $n \times m$  image block.

8. A system according to claim 6 wherein the processor conducts an inverse discrete cosine transform on the data according to the following:

$$B_{n \times m} = T_m^T \times C_{n \times m} \times T_n,$$

where  $B_{n \times m}$  denotes the inverse discrete cosine transformed image block with  $n$  pixels and  $m$  rows,  $T_n$  and  $T_m$  represent the horizontal and vertical integer transform matrices of size  $n \times n$  and  $m \times m$ , respectively, and  $C_{n \times m}$  denotes a cosine transformed  $n \times m$  image block.

9. A system according to claim 6 wherein the system is a device that receives, stores or transmits image data.
10. A system according to claim 6 including a memory that stores the transform matrix.
11. A system according to claim 9 wherein the memory stores different sized transform matrices, and the processor applies the different sized transform matrices according to a block size for a portion of the data being transformed.
12. A system according to claim 1 wherein the transform matrix is used for digital video coding.